Control Engineering Practice 94 (2020) 104195

Contents lists available at ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac

Fault detection and identification combining process measurements and statistical alarms ☆

Matthieu Lucke [a,b,*], Anna Stief [c], Moncef Chioua [a], James R. Ottewill [c], Nina F. Thornhill [b]

[a] ABB Corporate Research Germany, Wallstadter Strasse 59, 68526 Ladenburg, Germany
[b] Centre for Process Systems Engineering, Department of Chemical Engineering, Imperial College London, London SW7 2AZ, UK
[c] ABB Corporate Research Center, ul. Starowislna 13a, 31-038, Krakow, Poland

ARTICLE INFO

Keywords:
Fault detection and diagnosis
Variable selection

ABSTRACT

Classification-based methods for fault detection and identification can be difficult to implement in industrial systems where process measurements are subject to noise and to variability from one fault occurrence to another. This paper uses statistical alarms generated from process measurements to improve the robustness of the fault detection and identification on an industrial process. Two levels of alarms are defined according to the position of the alarm threshold: level-1 alarms (low severity threshold) and level-2 alarms (high severity threshold). Relevant variables are selected using the minimal-Redundancy-Maximal-Relevance criterion of level-2 alarms to only retain variables with large variations relative to the level of noise. The classification-based fault detection and identification fuses the results of a discrete Bayesian classifier on level-1 alarms and of a continuous Bayesian classifier on process measurements. The discrete classifier offers a practical way to deal with noise during the development of the fault, and the continuous classifier ensures a correct classification during later stages of the fault. The method is demonstrated on a multiphase flow facility.

1. Introduction

Fault detection determines whether a fault has happened, and fault identification determines the type of fault that occurred. In process literature, multivariate statistical process monitoring methods have been used for fault detection (Yin, Ding, Xie, & Luo, 2014), and contribution plot methods have been used as an extension of multivariate statistical process monitoring methods for finding the variables affected by the fault (fault isolation) (Alcala & Qin, 2009). By contrast, fault identification has been conducted using classification-based methods (Tong & Palazoglu, 2016).

Fault identification assumes that the types of faults, or classes of faults, are well-defined and have been encountered in the past so that historical fault occurrences can be used as a reference. Classification-based methods are trained to recognize patterns in the process measurements based on historical fault occurrences of the same class, where patterns refer to regularities in the data or in the characteristics of the data. In industrial systems, patterns in the process measurements for a given class of fault are subject to variability from one fault occurrence to another (Lucke, Chioua, Grimholt, Hollender, & Thornhill, 2018). This variability can be due to factors independent from the fault such as noise and external disturbances in the process measurements that are not related to the ongoing fault. The variability can also be due to the fault characteristics, for example with various fault amplitudes.

This paper presents a Fault Detection and Identification (FDI) method designed to deal with noise and with variability in the patterns of the process measurements. The method relies on statistical alarms generated from statistical processing of the process measurements. Two levels of alarms are defined according to the position of the alarm threshold: level-1 alarms (low severity threshold) and level-2 alarms (high severity threshold). The most informative variables with respect to the classification problem are selected using the minimal-Redundancy-Maximal-Relevance (mRMR) criterion (Peng, Long, & Ding, 2005). The mRMR criterion uses level-2 alarms instead of the continuous measurements in order to reduce the impact of small variations on the variable selection and to limit the effect of noise and external disturbances in the classification. The classification method for FDI also relies on statistical alarms, offering a Bayesian fusion of the outcome of a classifier based on level-1 alarms and the outcome of a classifier based on process measurements. The Bayesian classifier on level-1 alarms offers a good robustness to noise during the development of the fault. The Bayesian classifier on process measurements is used to identify the fault during later stages.

☆ This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skodowska-Curie grant agreement No 675215.

* Corresponding author at: ABB Corporate Research Germany, Wallstadter Strasse 59, 68526 Ladenburg, Germany.
E-mail address: matthieu.lucke@de.abb.com (M. Lucke).

https://doi.org/10.1016/j.conengprac.2019.104195
Received 30 April 2019; Received in revised form 14 October 2019; Accepted 15 October 2019
Available online xxxx

Section 2 provides additional information on statistical alarms and on FDI in the process literature. Section 3.1 presents the variable selection procedure based on the preliminary work by Lucke, Mei, Stief, Chioua, and Thornhill (2019), and Section 3.2 presents the FDI method with Bayesian fusion of process measurements and level-1 alarms. The benefits of integrating statistical alarms in FDI are illustrated on a simulation example in Section 4 and demonstrated on data from a multiphase flow facility in Sections 5 and 6.

2. Background

2.1. Statistical alarms

An alarm is an audible or visible means of indicating to the operator an equipment malfunction, process deviation, or abnormal condition requiring a timely response (IEC, 2014). Statistical alarms are a specific type of alarms generated based on statistical processing of a process measurement (IEC, 2014). A common statistical indicator used to generate statistical alarms is the standard deviation σ_i of each process variable X_i during normal operation in the historical data. A statistical alarm $\bar{X}_i$ is generated from a mean-centered process variable X_i as:

$$\bar{x}_i(t) = \begin{cases} -1, & \text{if} \quad x_i(t) \leq -\bar{k}\sigma_i \\ 0, & \text{if} \quad -\bar{k}\sigma_i < x_i(t) < \bar{k}\sigma_i \\ 1, & \text{if} \quad x_i(t) \geq \bar{k}\sigma_i \end{cases} \tag{1}$$

where $\bar{k}$ is a constant ($\bar{k} > 0$) that can be tuned for each alarm, and $\bar{k}\sigma_i$ is called the alarm threshold. In the following, the alarm $\bar{X}_i$ generated from X_i according to Eq. (1) with the constant $\bar{k}$ is described as level-1 alarm. A level-2 alarm $\bar{\bar{X}}_i$ is also generated from X_i according to Eq. (1) with the constant $\bar{\bar{k}}$ ($\bar{\bar{k}} > \bar{k} > 0$). $\bar{k}\sigma_i$ indicates a low severity threshold and $\bar{\bar{k}}\sigma_i$ indicates a high severity threshold. Commonly used values for $\bar{k}$ and $\bar{\bar{k}}$ are respectively 3 and 5, following statistical process control rules.

Alarm chattering (i.e. repeated transitions between the alarm state and the normal state in a short period of time) is usually removed from alarms to limit the nuisance to the operators (Naghoosi, Izadi, & Chen, 2011). Several methods have been designed to limit alarm chattering while maintaining a trade-off between false alarms and missed alarms, such as deadbands and delay timers (Adnan, Izadi, & Chen, 2011), and those methods can also be used on statistical alarms. In this paper, traditional delay-timers are applied to $\bar{X}_i$ and $\bar{\bar{X}}_i$, i.e. the alarm should cross the threshold for T_{on} consecutive samples before being activated (on-delay timer) and the alarm should go below the threshold for T_{off} consecutive samples before being deactivated (off-delay timer). T_{on} and T_{off} are tuned according to the dynamics of the corresponding variable.

The mean-centering of X_i is done offline using the mean in normal operation for each operating point. Fault identification for faults developing during transitions between operating points is considered as a separate research question that requires a classifier able to deal with highly non-stationary dynamics. In this work, it is assumed that one occurrence of each fault is available for training at one operating point and the fault detection and identification is applied on occurrences of the same faults at other operating points. Therefore, multi-mode alarming solutions are not included in the analysis.

2.2. Variable selection

2.2.1. Variable selection for fault detection and identification

FDI in industrial processes typically involves the analysis of a large number of process variables (Ming & Zhao, 2017). Variable selection and feature extraction are the two main approaches for reducing the number of variables, where a feature is defined as an individual measured property of the monitored system (Chandrashekar & Sahin, 2014). Variable selection chooses informative and discriminative variables for FDI (Ghosh, Ramteke, & Srinivasan, 2014), while feature extraction applies a transformation to the original variables to highlight characteristics or reduce the dimension (Ghosh et al., 2014). Historically, variable selection methods have not received the same attention as feature extraction methods in the process fault detection and diagnosis literature (Ming & Zhao, 2017). However, Ghosh et al. (2014) demonstrated that variable selection and feature extraction are complementary, and the recent review of Peres and Fogliatto (2018) highlights a growing interest in variable selection. The present article deals with joint fault detection and identification that assumes the types of faults are well-defined, so irrelevant variables can be eliminated through variable selection.

A popular strategy for variable selection combines pre-selection of variables with an optimization problem, where the set of variables leading to the best performance of the FDI algorithm is retained. The work reported by Verron, Tiplica, and Kobi (2008) combined a multivariate extension of the mutual information criterion with discriminant analysis for fault identification. Other criteria aiming at highlighting variables with abnormal variations have been suggested: (Zhao & Gao, 2017) identified the nonsteady faulty variables that are disturbed significantly using a stability factor, and Tong and Palazoglu (2016) used an index describing the degree of abnormal variation for each variable. Alternative approaches include genetic algorithms (Ghosh et al., 2014) or selection of the variables that correspond to the root causes of the faults (Shu, Ming, Cheng, Zhang, & Zhao, 2016). Regularization techniques (such as least absolute shrinkage and selection operators or elastic net) for variable selection in regression analysis have also been used for discriminant analysis in the context of fault isolation (Kuang, Yan, & Yao, 2015).

The variable selection method based on mutual information proposed by Verron et al. (2008) has become a benchmark in the literature. However, the analytical formulations of the univariate and multivariate mutual information of Verron et al. (2008) assume that the faults are stationary and that the process variables follow a Gaussian distribution during each fault. Faults are generally non-stationary in industrial systems and process variables evolve during the faults so that the Gaussian assumption does not apply. For this reason, a data-driven rather than analytical estimation of mutual information is preferred.

Multivariate data-driven estimation of mutual information is challenging because it assumes the estimation of joint probabilities that are heavy to compute. An alternative is to use pairwise comparisons taking into account a relevance criterion between each variable and the class, which assesses how the variables are related to the various classes of faults, and a redundancy criterion between each pair of variables that assesses how both variables are related. Ardakani et al. (2016) benchmarked multiple combinations of relevance criterion and redundancy criterion with different classifiers. The combination of a redundancy criterion and relevance criterion based on mutual information was introduced in Peng et al. (2005) as the minimal-Redundancy-Maximal-Relevance (mRMR) criterion. The next sub-section presents the direct application of the mRMR criterion to process measurements.

2.2.2. mRMR criterion for process measurements

The mutual information between two random variables quantifies the mutual dependence of the two variables (Shannon, 1948). In the context of variable selection, the simplest criterion is the univariate mutual information $I(X_i;C)$ between a continuous process variable X_i and the discrete class variable C. The higher the value of $I(X_i;C)$, the more relevant X_i is considered for the classification. It can be expressed as:

$$I(X_i;C) = \sum_{c\in C} \int_{X_i} P(x_i,c) \log \frac{P(x_i,c)}{P(x_i)P(c)} \, dx_i \tag{2}$$

where x_i and c represent the values that X_i and C can take. The probability distributions are computed using the extension of the nearest neighbors estimator between a continuous and a discrete variable (Ross, 2014).

The purpose of variable selection for classification is to find a set $\Omega_m = \{X_1, \dots, X_m\}$ of m variables X_i that have the largest dependency on the class C. In this paper, the values c of C corresponds to the fault classes, including normal operation. The max-dependency criterion is defined as:

$$\max_{\Omega_m} d(\Omega_m, C),\ d = I(\Omega_m; C) \tag{3}$$

where:

$$I(\Omega_m; C) = \sum_{c \in C} \int_{X_1} \dots \int_{X_m} P(x_1, \dots, x_m, c) \log \frac{P(x_1, \dots, x_m, c)}{P(x_1, \dots, x_m) P(c)}\ dx_1 \dots dx_m \tag{4}$$

Since the joint probability distributions $P(x_1, \dots, x_m, c)$ are difficult to estimate in practice, the max-dependency criterion is approximated using simplified criteria such as the mRMR criterion (Peng et al., 2005). The max-relevance criterion (Peng et al., 2005) is an approximation of the dependency criterion in Eq. (3) considering the mutual information between each variable X_i and the class C:

$$\max_{\Omega_m} D(\Omega_m, C),\ D(\Omega_m, C) = \frac{1}{|\Omega_m|} \sum_{X_i \in \Omega_m} I(X_i; C) \tag{5}$$

The min-redundancy criterion (Peng et al., 2005) considers the redundancy in the information of selected variables in a pairwise manner:

$$\min_{\Omega_m} R(\Omega_m),\ R(\Omega_m) = \frac{1}{|\Omega_m|^2} \sum_{X_i \in \Omega_m} \sum_{X_j \in \Omega_m} I(X_i; X_j) \tag{6}$$

While extending the redundancy analysis to more variables using multivariate mutual information is possible, it requires more data for the estimation and increases the computational load. Since the number of fault occurrences available for training are limited, the pairwise approximation of Peng et al. (2005) is preferred. The max-relevance criterion and the min-redundancy criterion are combined as the mRMR criterion:

$$\max_{\Omega_m} \Phi(D(\Omega_m, C), R(\Omega_m)),\ \Phi(D(\Omega_m, C), R(\Omega_m)) = D(\Omega_m, C) - R(\Omega_m) \tag{7}$$

$I(X_i; X_j)$ is computed using the nearest neighbors estimator explained in Kraskov, Stögbauer, and Grassberger (2004). $I(X_i; C)$ is computed using the extension of the nearest neighbors estimator between a continuous and a discrete variable described in Ross (2014).

In practice, an incremental search is performed to find the near-optimal set of variables defined by $\Phi(D(\Omega_m, C), R(\Omega_m))$. Assuming the set of $m-1$ variables Ω_{m-1} is known, the mth variable is selected from the set of remaining variables $\Omega - \Omega_{m-1}$ (where Ω indicates the set of all the variables X_i) as:

$$\max_{X_j \in \Omega - \Omega_{m-1}} \left[I(X_j; C) - \frac{1}{m-1} \sum_{X_i \in \Omega_{m-1}} I(X_j; X_i) \right] \tag{8}$$

2.3. Fault detection and identification

FDI is usually performed at each new sampling time t_0 using one (or several) classifier(s) based on the values $x_i(t_0)$ of the m process variables X_i in Ω_m. The observation vector $\mathbf{X}$ is defined as the m-dimensional vector of the process variables X_i, taking values $\mathbf{x}(t_0) = [x_1(t_0), x_2(t_0), \dots, x_m(t_0)]$ at time t_0. The classifier is trained considering the values of the observation vector $\mathbf{x}(t_n)$ at various times t_n ($n = 1 \,..\, N$, N being the total number of values used for training) and their corresponding class label $c(t_n)$ which relates either to a specific type of fault or to normal operation.

Classification-based methods assume that historical data about the faults are available, and new faults are detected and identified in a supervised manner (Tong & Palazoglu, 2016). Many types of classifiers have been proposed to identify faults from the process measurements, including discriminant analysis (Chiang, Kotanchek, & Kordon, 2004; He, Wang, Yang, & Yang, 2009; Zeng, Jia, Liang, & Gu, 2019), neural networks (Hoskins, Kaliyur, & Himmelblau, 1991; Wu & Zhao, 2018; Zhang & Zhao, 2017), support vector machines (Chiang et al., 2004), dissimilarity-based classifiers (Tong & Palazoglu, 2016), k-nearest neighbor classifiers (Zhu, Sun, & Romagnoli, 2018), and random forest classifiers (Chai & Zhao, 2019). Those methods have been designed to identify faults based on continuous features, i.e. based on the values of the process measurements, but the variability in the process measurements from one fault occurrence to another is not addressed, especially in simulated case studies. Traditional FDI methods also assume processes are stationary, but recent work addressed the issue of nonstationary fault characteristics through cointegration analysis (Hu & Zhao, 2019).

Unlike the classifiers cited above, Bayesian classifiers can integrate heterogeneous data. Bayesian methods for FDI have been introduced in the process literature through the work in Huang (2007) for control loop monitoring, which consists of detecting the under-performing control loops in the plant. The input data of the classifier were either discrete (e.g. values generated by control loop monitors) or continuous (e.g. through kernel density estimation of the probability density functions of the process measurements in Gonzalez and Huang (2014)). The work of Huang (2007) has been extended to FDI problems in specific cases where Bayesian methods bring advantages over traditional methods, e.g. for fault identification in multimode processes (Jiang, Huang, & Yan, 2016b) or for fault identification with heterogeneous data (Jiang, Huang, Ding, & Yan, 2016a; Stief, Ottewill, Tan, & Cao, 2018a). The fault are identified using a Bayesian classifier combining asynchronous measurements in Jiang et al. (2016a) or combining alarms from the alarm logs and process measurements in Stief et al. (2018a). A Bayesian framework was used to combine the classification outcomes of various classifiers for FDI in Tidriri, Tiplica, Chatti, and Verron (2018). Classification approaches using graphical models based on Bayesian networks have also been proposed for fault identification (Verron, Tiplica, & Kobi, 2006; Yu & Zhao, 2019).

The advantages that a Bayesian classification framework offers for integrating heterogeneous data, and in particular discrete and continuous data, are exploited in this paper for combining process measurements and statistical alarms.

2.4. Motivation for the method

Due to the limited number of fault occurrences available for training and to the variability of process measurements from one fault occurrence to another, classifiers should be made as robust as possible. This article suggests the use of statistical alarms to improve the robustness of the FDI to noise and variability in the process measurements.

Recent works highlighted the benefits of integrating alarms in FDI to improve the robustness of the methods. Stief et al. (2018a) combined alarms from alarm logs with process measurements using a two-stage Bayesian classifier to improve the accuracy of the classification by integrating a new source of information. Lucke et al. (2018) proposed a normalization of process measurements based on their alarm thresholds extracted from the alarm system to place the variations of the measurements in the context of their safe operating range.

Statistical alarms are generated directly from the process measurements. Statistical alarms have been used as a substitute for process measurements to reduce the computational load of methods based on probability density estimation, in particular for root cause analysis with transfer entropy (Hu, Wang, Chen, & Shah, 2017; Su et al., 2017; Yu & Yang, 2015). In the current work, statistical alarms are used as a substitute for process measurements in variable selection to limit the impact of the small variations in the estimation of the mutual information. Statistical alarms are also used in combination with process measurements for FDI using a two-stage Bayesian classification approach similar to the one in Stief et al. (2018a). However, while Stief et al. (2018a) extracted only discrete features from the process measurements, the present article suggests a combination of discrete and

continuous features. Statistical alarms are used as features for the detection and identification during the early stage of the fault since they are designed to indicate deviations from the normal operation while limiting the number of transitions due to chattering. The values of the process measurements are used to provide a more accurate fault identification during the late stage of the fault.

The Bayesian framework offers a simple approach to integrate continuous and discrete features in the classification. A discrete Bayesian classifier gives probabilities of the classification outcome based on the values of the alarms after estimation of the discrete distributions on the training set. A continuous Bayesian classifier gives probabilities of the classification outcome based on the values of the process measurements after estimation of the probability density functions using kernel density estimation on the training set. The final decision is taken based on a fusion of the outcomes of both classifiers.

3. Method

3.1. Variable selection with mRMR on level-2 alarms

In order to reduce the impact of small variations on the variable selection, and to limit the effects of noise and external disturbances on the classification, the mRMR criterion (Peng et al., 2005) is applied to level-2 alarms $\bar{\bar{X}}_i$ (i.e. the statistical alarms with high severity threshold $\bar{\bar{k}}\sigma_i$) rather than to the original process measurements. The main difference with the preliminary work in Lucke et al. (2019) is that the classification accuracies for the various sets of variables are computed with the Bayesian classifier presented in the next section. The accuracy of the classifier is defined as the proportion of values $\mathbf{x}(t_0)$ of the observation vector $\mathbf{X}$ that are correctly classified. The mutual information $I(\bar{\bar{X}}_i; C)$ based on $\bar{\bar{X}}_i$ can be computed as a discrete sum:

$$I(\bar{\bar{X}}_i; C) = \sum_{c\in C}\sum_{\bar{\bar{x}}_i\in\bar{\bar{X}}_i} P(\bar{\bar{x}}_i, c)\log\frac{P(\bar{\bar{x}}_i, c)}{P(\bar{\bar{x}}_i)P(c)} \tag{9}$$

The incremental search of Section 2.2.2 for the mRMR criterion based on the level-2 alarms $\bar{\bar{X}}_i$ becomes:

$$\max_{\bar{\bar{X}}_j\in\bar{\bar{\Omega}}-\bar{\bar{\Omega}}_{m-1}}\left[I(\bar{\bar{X}}_j; C) - \frac{1}{m-1}\sum_{\bar{\bar{X}}_i\in\bar{\bar{\Omega}}_{m-1}} I(\bar{\bar{X}}_j; \bar{\bar{X}}_i)\right] \tag{10}$$

where $\bar{\bar{\Omega}}_{m-1}$ indicates the set of the best $m-1$ variables $\bar{\bar{X}}_i$ and $\bar{\bar{\Omega}}$ the set of all variables $\bar{\bar{X}}_i$. Both $I(\bar{\bar{X}}_i; C)$ and $I(\bar{\bar{X}}_i; \bar{\bar{X}}_j)$ are computed as discrete sums.

3.2. FDI with Bayesian fusion of process measurements and level-1 alarms

3.2.1. Bayesian classification with level-1 alarms

The discrete observation vector $\bar{\mathbf{X}}$ is defined as the vector of the level-1 alarms $\bar{X}_i$ (i.e. the statistical alarms with low severity threshold $\bar{k}\sigma_i$) associated with the m selected variables $X_i \in \Omega_m$, taking value $\bar{\mathbf{x}}(t_0) = [\bar{x}_1(t_0), \bar{x}_2(t_0), \dots, \bar{x}_m(t_0)]$ at time t_0. The probability that a value $\bar{\mathbf{x}}(t_0)$ is associated with class c is computed as:

$$P(C = c|\bar{\mathbf{X}} = \bar{\mathbf{x}}(t_0)) = \frac{P(\bar{\mathbf{x}}(t_0)|c)}{P(\bar{\mathbf{x}}(t_0))}P(c) = \frac{P(\bar{\mathbf{x}}(t_0)|c)}{\sum_{c'\in C}P(\bar{\mathbf{x}}(t_0)|c')}P(c) \tag{11}$$

where $P(c)$ is the prior probability of class c, $P(c|\bar{\mathbf{x}}(t_0))$ is the posterior probability of class c, $P(\bar{\mathbf{x}}(t_0)|c)$ is the likelihood, and $P(\bar{\mathbf{x}}(t_0))$ is the probability of observing the value $\bar{\mathbf{x}}(t_0)$. Since all of the values are discrete, the probabilities are computed as discrete sums on the training data.

3.2.2. Bayesian classification with process measurements

The continuous observation vector $\mathbf{X}$ is defined as the vector of the m selected variables $X_i \in \Omega_m$, taking value $\mathbf{x}(t_0) = [x_1(t_0), x_2(t_0), \dots, x_m(t_0)]$ at time t_0. The probability that a given value $\mathbf{x}(t_0)$ is associated with class c can be computed using Bayes' rule as:

$$P(C = c|\mathbf{X} = \mathbf{x}(t_0)) = \frac{P(\mathbf{x}(t_0)|c)}{P(\mathbf{x}(t_0))}P(c) \tag{12}$$

where $P(c)$ is the prior probability of class c, $P(c|\mathbf{x}(t_0))$ is the posterior probability of class c (i.e. once the value $\mathbf{x}(t_0)$ at time t_0 has been observed), $P(\mathbf{x}(t_0)|c)$ is the likelihood (i.e. the probability of the value $\mathbf{x}(t_0)$ assuming class c), and $P(\mathbf{x}(t_0))$ is the probability of observing the value $\mathbf{x}(t_0)$.

The probability density function $f(\mathbf{x}(t_0))$ of $\mathbf{X}$ is estimated using kernel density estimation (Parzen, 2007). The estimated probability density function $\hat{f}(\mathbf{x}(t_0), \mathbf{H})$ is a summation of kernel functions centered around the values $\mathbf{x}(t_n)$ of the observation vectors used for training ($n = 1 \,..\, N$, N being the number of values used for training):

$$\hat{f}(\mathbf{x}(t_0), \mathbf{H}) = \frac{1}{N}\sum_{n=1}^{N}|\mathbf{H}|^{-\frac{1}{2}}K\left(\mathbf{H}^{-\frac{1}{2}}(\mathbf{x}(t_0) - \mathbf{x}(t_n))\right) \tag{13}$$

$\mathbf{H}$ is a symmetric positive definite matrix called the bandwidth matrix, K is a m-variate kernel function such that $\int K(\mathbf{x})d\mathbf{x} = 1$. A popular choice for the kernel function is the m-variate normal density function:

$$K(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{m}{2}}}\exp(-\frac{1}{2}\mathbf{x}^T\mathbf{x}) \tag{14}$$

which gives the kernel density estimate:

$$\hat{f}(\mathbf{x}(t_0), \mathbf{H}) = \frac{1}{N\sqrt{(2\pi)^m|\mathbf{H}|}}\sum_{n=1}^{N}\exp\left(-\frac{1}{2}(\mathbf{x}(t_0) - \mathbf{x}(t_n))^T\mathbf{H}^{-1}(\mathbf{x}(t_0) - \mathbf{x}(t_n))\right) \tag{15}$$

The ratio $\frac{P(\mathbf{x}(t_0)|c)}{P(\mathbf{x}(t_0))}$ of Eq. (12) can thus be expressed with the kernel estimates of the probability densities $\hat{f}(\mathbf{x}(t_0)|c)$ and $\hat{f}(\mathbf{x}(t_0))$[1]:

$$P(C = c|\mathbf{X} = \mathbf{x}(t_0)) = \frac{\hat{f}(\mathbf{x}(t_0)|c)}{\hat{f}(\mathbf{x}(t_0))}P(c) = \frac{\hat{f}(\mathbf{x}(t_0)|c)}{\sum_{c'\in C}\hat{f}(\mathbf{x}(t_0)|c')}P(c) \tag{16}$$

In this paper, each probability density function $\hat{f}(\mathbf{x}(t_0)|c)$ corresponding to a class c is estimated using the bandwidth matrix $\mathbf{H}(c)$ defined as the diagonal matrix of $h(c)\sigma_i$:

$$\mathbf{H}(c) = h(c)\begin{bmatrix}\sigma_1 & & \\ & \ddots & \\ & & \sigma_m\end{bmatrix} \tag{17}$$

where $h(c)$ is a positive constant tuned for each class c (Section 6.3.3) and σ_i is the standard deviation during normal operation data for the variable X_i.

3.2.3. Bayesian fusion

The final classification is taken based on the decision level fusion of the discrete and the continuous classifiers. The predicted class c_f for a given value $\mathbf{x}(t_0)$ of the observation vector $\mathbf{X}$ and a given value $\bar{\mathbf{x}}(t_0)$ of the corresponding discrete observation vector $\bar{\mathbf{X}}$ is computed merging the posterior probabilities $P(C = c|\bar{\mathbf{X}} = \bar{\mathbf{x}}(t_0))$ of the discrete classifier and the posterior probabilities $P(C = c|\mathbf{X} = \mathbf{x}(t_0))$ of the continuous classifier:

$$\begin{aligned} c_f &= \arg\max_{c\in C}\left(P(c|\mathbf{x}(t_0))P(c|\bar{\mathbf{x}}(t_0))\right) \\ &= \arg\max_{c\in C}\left(\frac{\hat{f}(\mathbf{x}(t_0)|c)}{\sum_{c'\in C}\hat{f}(\mathbf{x}(t_0)|c')}P(c)\frac{P(\bar{\mathbf{x}}(t_0)|c)}{\sum_{c'\in C}P(\bar{\mathbf{x}}(t_0)|c')}P(c)\right) \end{aligned} \tag{18}$$

[1] As explained in John and Langley (1995), Eq. (16) is not strictly correct. The probability that a continuous-valued random variable exactly equals any value is zero. Instead the variable lies within an interval of size Δ, and for a very small constant Δ, $P(\mathbf{x}(t_0)) \approx f(\mathbf{x}(t_0))\Delta$. The Δ factors cancel out in the numerator and denominator of Eq. (12).

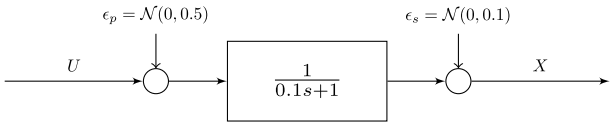

Fig. 1. System of the simulation example.

In this case, the prior probability $P(c)$ is chosen as uniform (i.e. all fault classes c have the same prior probabilities $P(c)$) so the expression can be simplified:

$$c_f = \arg\max_{c \in C} \left(\frac{\hat{f}(\mathbf{x}(t_0)|c)}{\sum_{c' \in C} \hat{f}(\mathbf{x}(t_0)|c')} \frac{P(\bar{\mathbf{x}}(t_0)|c)}{\sum_{c' \in C} P(\bar{\mathbf{x}}(t_0)|c')} \right) \tag{19}$$

4. Tests of the method in simulation

4.1. Simulation example

The benefits of integrating type-1 alarms in the classification are demonstrated in a simple case study. The system is a first order systems with input U and output X, described in Fig. 1. Process noise ϵ_p and sensor noise ϵ_p are added according to Fig. 1. Type-1 alarms $\bar{X}$ are generated from X according to Eq. (1) with $\bar{k} = 3$. T_{on} is set to 5 for the on-delay timer and T_{off} is set to 10 for the off-delay timer.

Two types of faults are introduced as input U of the system after 100 samples of normal operation and the fault detection and identification is done based on the output X of the system. Two occurrences of each type of fault with various amplitudes are generated (one for training the classifier and one for testing) and inserted as input U of the system. The two faults are:

- Fault 1 (F1): U is the step response of a second order system $G(s) = \frac{1}{20s^2+0.5s+1}$. The training occurrence is generated with a step of amplitude 20, and the test occurrence is generated with a step of amplitude 15.
- Fault 2 (F2): U is a step of amplitude 3 for the training occurrence and amplitude 2 for the test occurrence.
- Normal operation (N0): all samples before the introduction of the faults constitute the normal operation data.

4.2. Results

Fig. 2 summarizes the results. The plots in the first row shows the outputs X during the test occurrence of each fault and the corresponding alarms $\bar{X}$ in red. The vertical line indicates the beginning of the fault after 100 samples. The following plots show respectively the classification outcome c_c with the continuous Bayesian classifier, the classification outcome c_d with the discrete Bayesian classifier, and the classification outcome c_f with the fusion of both classifiers. The plots in the last row show the classification outcome c_L of a Linear Discriminant Analysis (LDA) classifier on the process measurements for comparison. LDA is a traditional method for FDI on process measurements (Chiang et al., 2004) but is not adapted to deal with discrete features, like most FDI methods in the literature. The benefits of integrating statistical alarms in FDI are demonstrated through a comparison with a traditional FDI method on process measurements only. The accuracies of the classifiers are listed in Table 1.

4.3. Comparison of the methods

The classification outcome of the continuous Bayesian classifier in Fig. 2 shows that the continuous classifier cannot distinguish clearly between normal operation N0 and F1 due to the noisy behavior of X during fault F1. When it comes to fault F1, the alarm $\bar{X}$ (in red) offers a better separation between normal operation and the fault than X.

Table 1
Classification accuracies on the simulation example.

	Accuracy
Continuous	0.934
Discrete	0.594
Fusion	**0.991**
LDA	0.920

The on-delay and off-delay timers limit the number of transitions in the state of the alarm and $\bar{X}$ shows a clear transition between N0 and F1 once the threshold has been crossed for T_{on} consecutive samples. Thus, the discrete classifier offers a better distinction between normal operation and the faults, although it is not able to distinguish F1 from F2 since it classifies both faults as F2.

The fusion of both classifiers offers both a clear distinction between normal operation and the faults, and between F1 and F2. The delay in the alarm activation introduced by the on-delay timer has a limited impact on the detection of F1 and F2 with the fusion of both classifiers: the jump in value of X during F1 and F2 gives a high fault probability in the continuous classifier, and the outcome of the fusion of both classifiers indicates a fault even before the alarm activates.

As a comparison, the outcome of the LDA classifier in Fig. 2 presents the same characteristics as the outcome of the continuous Bayesian classifier. The LDA classifier is unable to distinguish clearly F1 from normal operation.

5. Application to industrial case study

5.1. The multiphase flow facility

The case study is a multiphase flow facility located at the Process System Engineering laboratory of Cranfield University described by Stief, Tan, Cao, and Ottewill (2018b) and depicted in Fig. 3. Air (the green stream) and water (the blue stream) are mixed just below PT417 in the top part of the plant, and the mixed two-phase flow then goes through the horizontal section in the center of the figure. The three-phase separator at the bottom of the figure separates the mixed flow into water and air, the water is reinjected in the system through a water coalescer, and some air is exhausted in the atmosphere though VC501.

The system is operated at two different points and normal operation data are gathered for each operating point. Operating point A corresponds to an air flow rate of 120 $sm^3\ h^{-1}$ and a water flow rate of 0.1 $kg\ s^{-1}$. Operating point B corresponds to an air flow rate of 150 $sm^3\ h^{-1}$ and a water flow rate of 0.5 $kg\ s^{-1}$. Three types of fault were induced successively at each operating point:

- Air blockage (F1): valve V11 was gradually closed to simulate a developing blockage in the input airline.
- Air leakage (F2): valve V10 was gradually opened so that the air is partially leaked out to the atmosphere.
- Diverted flow (F3): bypass valve U39 was gradually opened so that the mixed flow is partially led straight to the riser and partially led into the horizontal pipeline before joining the riser.
- Normal operation (N0): all samples before the introduction of the faults constitute the normal operation data.

The data are separated into four classes, the normal operation data and the three faulty episodes. The analysis focuses on the 17 process variables listed in Table 2. The variables are mean-centered at each operating point. $\bar{X}_i$ and $\bar{\bar{X}}_i$ for operating point A are generated using σ_i in operating point A, and $\bar{X}_i$ and $\bar{\bar{X}}_i$ for operating point B are generated using σ_i in operating point B. According to Section 2.1, $\bar{k}$ is set to 3 for the generation of level-1 alarms and $\bar{\bar{k}}$ is set to 5 for the generation of level-2 alarms. T_{on} is set to 30 for the on-delay timer and T_{off} is set to 60 for the off-delay timer.

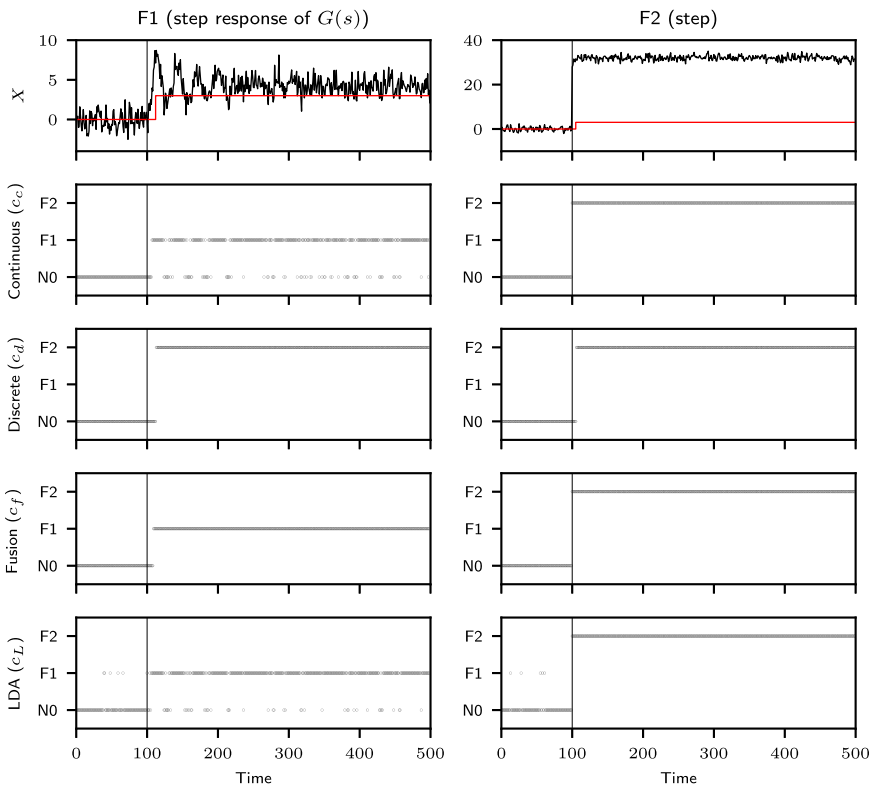

Fig. 2. Standardized measurements, level-1 alarms (in red), and classification outcomes with continuous Bayesian classifier, discrete Bayesian classifier, fusion of both Bayesian classifiers, and LDA on the simulation example. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

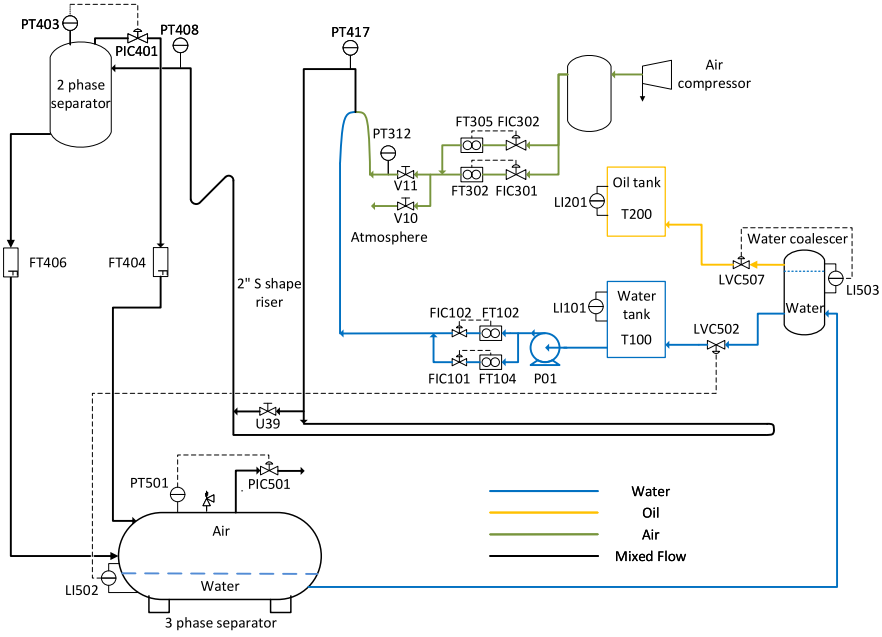

Fig. 3. Process diagram of the multiphase flow facility reproduced from Stief et al. (2018b).[2] (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.2. Design of experiment

Variable selection and training of the classification algorithm are performed on the data of one operating point, and tested on the data of the other operating point. Two scenarios are presented: training on operating point A and testing on operating point B (scenario AB), and training on operating point B and testing on operating point A (scenario BA).

The variable ranking is performed on a training dataset containing the values $\mathbf{x}(t_n)$ of the observation vector $\mathbf{X}$ during one occurrence of each fault and during normal operation. The variable ranking is computed using the mRMR criterion on the level-2 alarms. As a comparison,

[2] ©2018 International Federation of Automatic Control. Reproduced with permission from the original publication in IFAC-PapersOnline, 51/18.

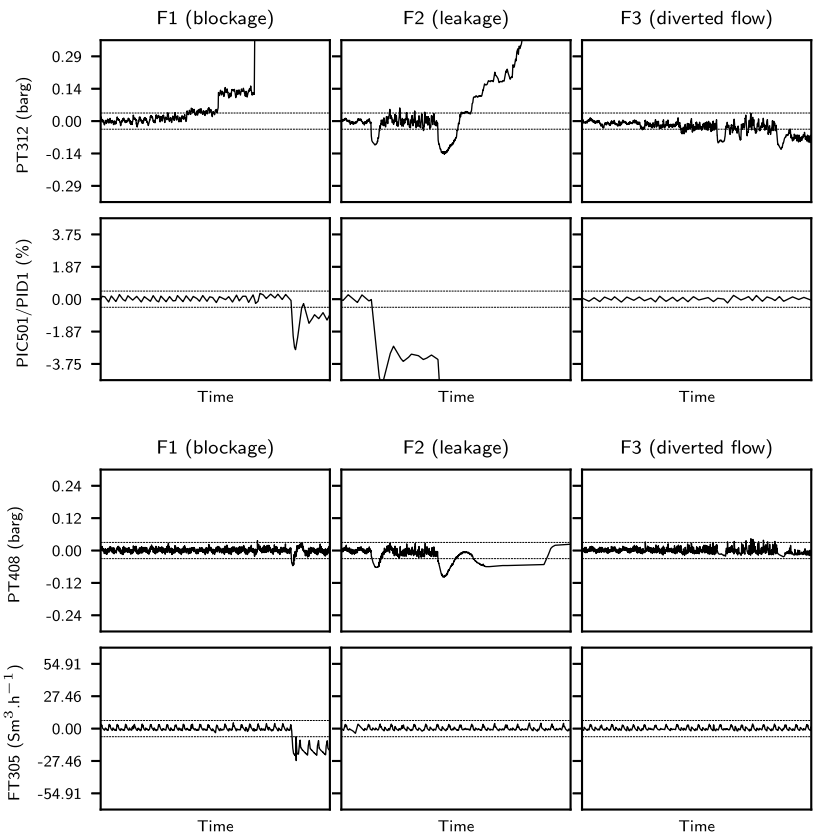

Fig. 4. First two ranked mean-centered process variables with mRMR ranking on level-2 alarms (top) and with mRMR ranking on process measurements in operating point A (scenario AB). The dotted lines correspond to the level-2 alarm thresholds ($\bar{\bar{k}} = 5$).

another variable ranking is computed using the mRMR criterion on process measurements (Section 2.2.2).

All the sets Ω_m corresponding to the m best ranked variables ($m = 1 \,..\, M$ with M the total number of variables) are assessed for each variable ranking. The accuracy of the classifier is computed on the same training dataset using a three-fold cross-validation strategy. The training dataset is divided in three parts that are used successively as test set while the two remaining parts are used for training, and the final accuracy represents the average accuracy of the three tests. The classifier chosen for the experiment is the fusion of the continuous classifier and the discrete classifier from Section 3.2.3. Finally, the robustness of each variable ranking is evaluated on a test dataset containing another occurrence of each fault and new normal operation data. The classification accuracy for each variable set Ω_m is computed by training the classifier on the training dataset and testing it on the test dataset.

6. Results

6.1. Variable selection

This section illustrates how variable selection on level-2 alarms prioritizes variables with large variations compared to variable selection on the process measurements, and how it directly impacts the robustness of the classification.

6.1.1. Variable ranking

Fig. 4 shows the two top ranked mean-centered variables in operating point A with mRMR on the level-2 alarms and with mRMR on the process measurements, both in scenario AB. The dotted lines indicate the level-2 alarm thresholds with $\bar{\bar{k}} = 5$. Each column depicts one of the faults. While mRMR on the level-2 alarms prioritizes variables with large variations, mRMR on the continuous measurements picks some variables with variations that stay close to the normal range. Therefore, noise and external disturbances have a stronger effect on the patterns of the measurements of the bottom plot than on the patterns of the measurements of the top plot.

Table 2
Process variables from the multiphase flow facility.

Sensor tag	Process variable
FT305/OUT	Inlet air flow rate 1
FT302/OUT	Inlet air flow rate 2
PT312/OUT	Air delivery pressure
PT417/OUT	Pressure in the mixing zone
PT408/OUT	Pressure at the riser top
PT403/OUT	Pressure in the top separator
FT404/OUT	Top separator output air flow rate
FT406/OUT	Top separator output water flow rate
PT501/OUT	Pressure in the three-phase separator
PIC501/PID1/OUT	Air outlet valve opening in the three-phase separator
LI502/OUT	Water-oil level three-phase separator
LI503/OUT	Water coalescer level
LVC502/PID1/OUT	Water coalescer outlet valve opening
LI101/OUT	Water tank Level
FIC302/PID1/OUT	Inlet air flow rate controller 1 valve opening
FIC302/PID1/PV	Inlet air flow rate controller 1 process value
FIC301/PID1/PV	Inlet air flow rate controller 2 process value

6.1.2. Classification accuracies of the sets of variables

Fig. 5 shows the classification accuracies obtained for the training data with cross-validation (solid lines) and those obtained for the test data (dotted lines) for the various sets Ω_m of the m best ranked variables, with m from 1 to 10. The top plot corresponds to scenario AB, the bottom plot to scenario BA.

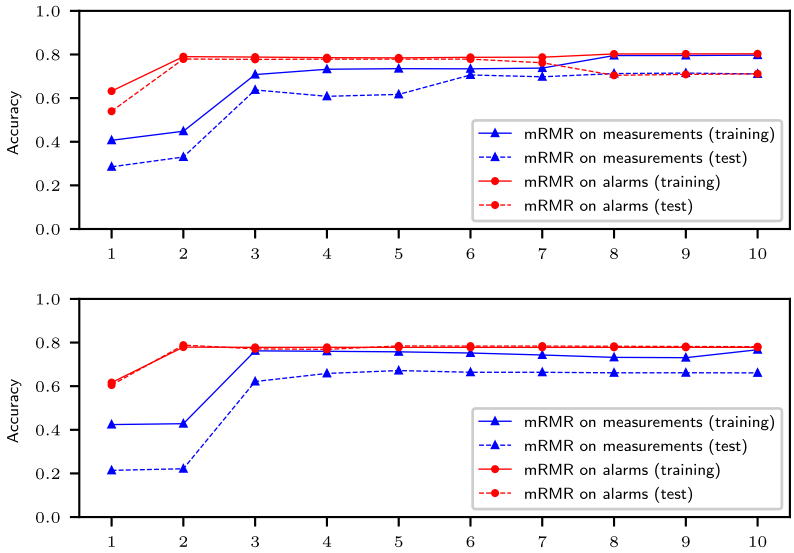

Fig. 5. Classification accuracy on training and test data for scenario AB (top) and scenario BA (bottom) for the first 10 variables with mRMR ranking applied to process measurements (blue) and level-2 alarms (red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Fig. 5 highlights the contrast between classification accuracy with mRMR ranking on level-2 alarms (red lines) and classification accuracy with mRMR ranking on process measurements (blue lines). The classification accuracy when using level-2 alarms increases quickly for the first ranked variables and becomes flat or starts decreasing after a certain number of variables (in this case, two variables). By contrast, the classification accuracy when using process measurements increases slowly for the first variables, which means that the first ranked variables with mRMR on process measurements are not able to discriminate between the different classes as the variables selected with mRMR on level-2 alarms. In addition, the difference between the accuracy on the training data and the accuracy on the test data is larger when using mRMR on the process measurements than when using mRMR on the alarms. This shows that there is overfitting in the classification with variables selected with mRMR on the process measurements, and that the corresponding classifier is not robust.

6.1.3. Selected set of variables

The set Ω_2 of the two highest-ranked variables with mRMR ranking on the level-2 alarms is selected as best set of variables based on the classification accuracy on the training and test data in both scenarios AB and BA. Ω_2 is used for the FDI in the next sections.

The two variables in Ω_2 are relevant with regard to the considered faults. The highest-ranked variable, PT312, corresponds to the air delivery pressure, measured just after the valves V10 and V11 that are used to induce the blockage and leakage faults, and just before the bypass valve U39 used to induce the diverted flow fault. The second variable, PIC501/PID1, refers to the air outlet valve of the three-phase separator. As the controller output of the valve opening, PIC501/PID1 gives an indication of the value of the outlet air flow, which is not directly measured in the facility. The blockage and the leakage fault lead to a reduction in the quantity of air introduced in the mixed flow, which triggers a closure of the air outlet valve of the three-phase separator to maintain the air pressure of the three-phase separator at its setpoint.

6.2. Fault detection and identification

This section investigates Bayesian classification of the selected variables, based on fusion of a continuous classifier on process measurements and a discrete classifier on the corresponding level-1 alarms.

Table 3
Classification accuracies for scenarios AB and BA.

	Scenario AB	Scenario BA
Continuous	0.678	0.72
Discrete	0.719	0.676
Fusion	**0.779**	**0.788**
QDA	0.702	0.740

Table 4
Confusion matrices for scenarios AB and BA.

		Scenario AB				Scenario BA			
		N0	F1	F2	F3	N0	F1	F2	F3
Continuous	N0	**0.94**	0.00	0.00	0.06	**0.99**	0.00	0.00	0.01
	F1	0.50	**0.50**	0.00	0.01	0.38	**0.60**	0.00	0.02
	F2	0.09	0.30	**0.61**	0.01	0.13	0.00	**0.86**	0.01
	F3	0.15	0.01	0.00	**0.84**	0.51	0.00	0.00	**0.49**
Discrete	N0	**1.00**	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00
	F1	0.43	**0.43**	0.14	0.00	0.38	**0.62**	0.00	0.00
	F2	0.11	0.00	**0.89**	0.00	0.14	0.47	**0.39**	0.00
	F3	0.18	0.00	0.00	**0.82**	0.27	0.00	0.00	**0.73**
Fusion	N0	**1.00**	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00
	F1	0.43	**0.57**	0.00	0.00	0.38	**0.62**	0.00	0.00
	F2	0.09	0.01	**0.89**	0.00	0.13	0.01	**0.85**	0.01
	F3	0.15	0.00	0.00	**0.85**	0.26	0.00	0.00	**0.74**
QDA	N0	**0.95**	0.00	0.00	0.05	**0.97**	0.00	0.00	0.03
	F1	0.37	**0.41**	0.04	0.18	0.33	**0.56**	0.00	0.11
	F2	0.09	0.01	**0.90**	0.00	0.13	0.00	**0.87**	0.00
	F3	0.15	0.04	0.00	**0.80**	0.36	0.00	0.00	**0.64**

6.2.1. Classification results

Fig. 6 summarizes the classification outcome for scenario AB. The plots of the first two rows show the standardized selected variables (in black) and the corresponding level-1 alarms (in red) for faults F1, F2, and F3. The vertical line indicates the beginning of the fault: the samples before the vertical line correspond to normal operation (N0). The bottom plots show respectively the classification outcome c_c with the continuous Bayesian classifier, the classification outcome c_d with the discrete Bayesian classifier, and the classification outcome c_f with the fusion of both classifiers. The plots in the last row show the classification outcome c_Q of a Quadratic Discriminant Analysis (QDA) classifier on the process measurements for comparison. Quadratic discriminant analysis is the quadratic extension of linear discriminant analysis used in Section 4, which could not be used because the faults

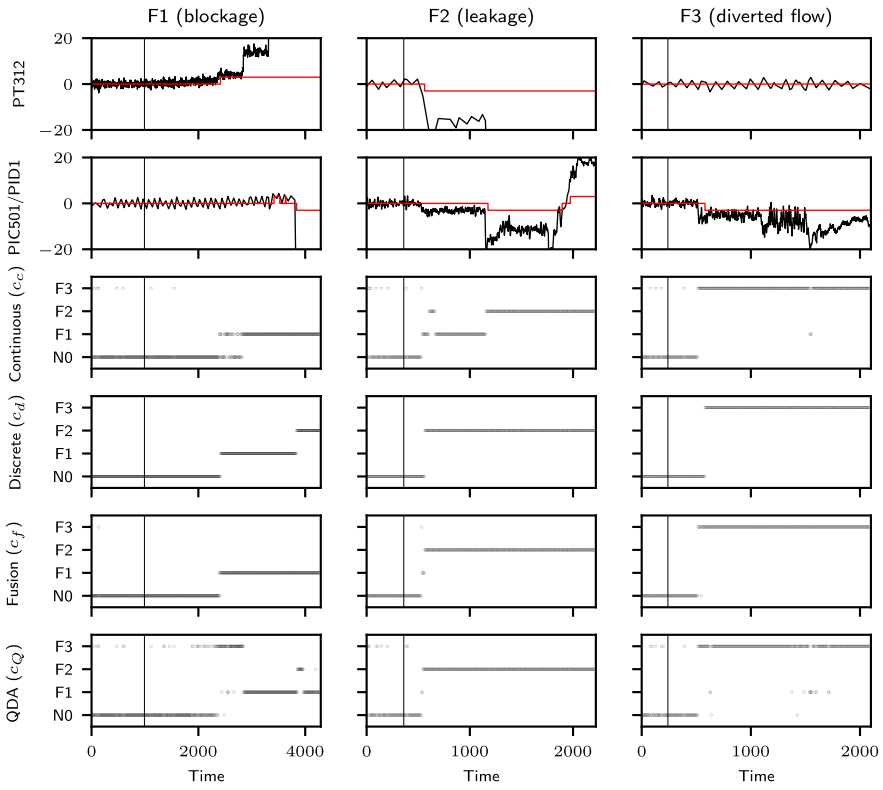

Fig. 6. Standardized measurements, level-1 alarms (in red), and classification outcomes with continuous Bayesian classifier, discrete Bayesian classifier, fusion of both Bayesian classifiers, and QDA (scenario AB). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

are not linearly separable in this case. As in Section 4, QDA is applied on the process measurements only, since it is not adapted to discrete features. Fig. 7 summarizes the classification outcome for scenario BA.

The classification accuracies corresponding to Figs. 6 and 7 are indicated in Table 3. The accuracy indicates the proportion of values of the observation vector correctly classified. The confusion matrices are presented in Table 4. The figures indicate the proportion of values of the observation vector classified as each class. The columns correspond to the predicted classes and the rows correspond to the actual classes.

6.2.2. Comparison of the methods

Both Figs. 6 and 7 show that the continuous Bayesian classifier is able to classify all faults correctly after a certain time. However, the detection occurs late and the classifier has troubles clearly distinguishing the faults from normal operation during their development, in particular for F3. On the opposite, the discrete Bayesian classifier shows a clear and early detection of the faults due to the activation of the alarms, but may not be able to distinguish between faults during the late stage of the fault when all the alarms have reached the same values. For example, the discrete classifier classifies F1 as F2 in Fig. 6 or F2 as F1 in Fig. 7 at the end of the faults because the alarms have reached the same values in F1 and in F2.

The fusion of both classifiers brings together the advantages of both the continuous and the discrete classifiers. The discrete classifier ensures a clear detection and identification of the fault as soon an alarm gets activated, and the continuous classifier ensures a correct classification when the values of the alarms lead to indetermination.

In comparison, the QDA classifier on the process measurements presents the same characteristics as the continuous Bayesian classifier. Although the QDA classifier classifies all faults correctly, it cannot clearly separates fault classes when the level of noise is high compared to the pattern of the fault. This is the case for F3 which is regularly classified as normal, or F1 which is regularly classified as F3. In addition, the QDA classifier classifies several samples from normal operation as F3, which leads to false detections. Limiting the number of false detections is another advantage of the discrete Bayesian classifier, which explains why the fusion of the discrete and continuous classifiers presents very few false detections in scenario AB and BA.

6.3. Analysis of the parameters

6.3.1. Alarm thresholds

The role of level-1 alarms is to indicate any deviation from the normal operating range. For this reason, the objectives for designing level-1 alarms and for designing a fault detection system are similar, and the level-1 alarms can directly be used for fault detection. It is common practice to set the alarm threshold to $3\sigma_i$ (σ_i indicates the standard deviation in normal operation) following basic statistical process control rules for a Gaussian variable. In this paper, the variables are assumed to be Gaussian during normal operation and the $3\sigma_i$ threshold is retained.

The role of level-2 alarms is to indicate a severe deviation from the normal operating range. In this paper, level-2 alarms are used to select only the variable with severe deviations from the normal operating range. The variables whose level-2 alarms can best discriminate fault classes are good candidates for a robust classification, since the effect of noise and external disturbances compared to the fault pattern is limited on those variables. This variable selection approach explains why the level-1 alarms are also helpful, to some extent, in discriminating fault classes for the fault identification. Similarly to the level-1 alarm threshold, the level-2 alarm threshold can be set as a multiple of σ_i ($5\sigma_i$ is common practice).

6.3.2. Delay timers

Delay timers are set to reduced the number of alarm transitions not to overload the operators. T_{on} and T_{off} for the on-delay and off-delay timers are set during the alarm design stage along with the alarm thresholds. From a safety perspective, the on-delay timer has a higher

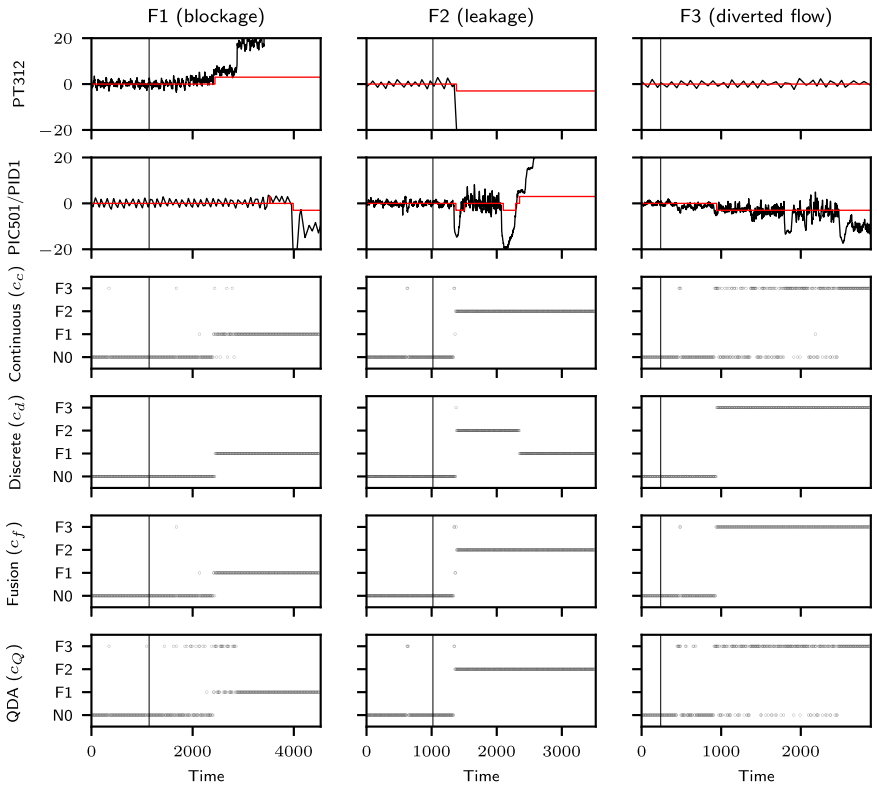

Fig. 7. Standardized measurements, level-1 alarms (in red), and classification outcomes with continuous Bayesian classifier, discrete Bayesian classifier, fusion of both Bayesian classifiers, and QDA (scenario BA). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

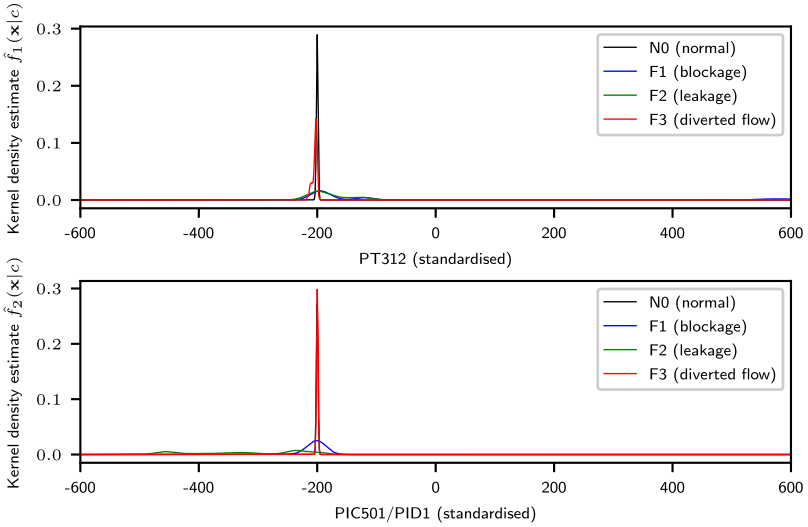

Fig. 8. Kernel density estimates of the probability density functions of the standardized variables for normal operation N0 and faults F1, F2, and F3 in operating point A.

impact than the off-delay timer since it delays the activation of the alarm (Adnan et al., 2011), which can be an issue in case of real fault. For this reasons, T_{on} is usually taken smaller than T_{off}. In this paper, $T_{off} = 2T_{on}$ for both the simulation and the industrial case study. T_{on} and T_{off} also depends on the dynamics of the process variable. For the industrial case study, an on-delay timer of 30 s and an off-delay timer of 60 s are used, which correspond to standard industrial values.

6.3.3. Kernel bandwidth for the continuous Bayesian classifier

The continuous Bayesian classifier assumes the estimation of the probability density functions of the selected variables (Eq. (16)). The estimated probability density functions are used to assign the probability of each fault class for each new observation vector. In this paper, the constant $h(c)$ in the bandwidth matrix $\mathbf{H}(\mathbf{c})$ (Eq. (17)) is used to adjust the smoothness of the estimated probability density functions.

A common method for choosing a good value of the kernel bandwidth in kernel density estimation is cross-validation (Wand, Jones, & Jones, 1994). In this paper, a value of $h(c)$ is determined for each fault class c using three-fold cross-validation on the training dataset corresponding to fault c. The training dataset corresponding to fault c is divided in three parts that are used successively to test the fit of the kernel density estimate while the two remaining parts are used for computing the kernel density estimate. The fit of the kernel density estimate is computed using the log likelihood of the data that was not used for computing the kernel density estimate, and the value of $h(c)$

maximizing the log likelihood is chosen for the estimation of $\hat{f}(\mathbf{x}|c)$ in the continuous classifier.

The kernel density estimates $\hat{f}_1(\mathbf{x}|c)$ and $\hat{f}_2(\mathbf{x}|c)$ of the probability density functions of the two selected standardized variables are shown in Fig. 8 for each fault occurrence. In this case, h is set to 1 for N0 and F3, and to 15 for F2 and F3.

7. Conclusion

Classification-based methods are popular for fault detection and identification but their implementation in the industry is limited by noise and variability of process measurements from one fault occurrence to another. This paper demonstrated how statistical alarms could be used to improve the robustness of the fault detection and identification. A variable selection stage identifies the most relevant variables for the classification based on the corresponding statistical alarms with high severity threshold, focusing on variables with large variations where the impact of noise and external disturbances is limited. The classification stage uses a Bayesian framework combining the outcome of a discrete classifier based on statistical alarms with low severity threshold and the outcome of a continuous classifier based on process measurements. The discrete classifier presents a good robustness to noise for fault detection and identification during the early stage of the fault since the number of alarm transitions are limited by the delay timers. The continuous classifier ensures that the fault is identified correctly during later stages of the fault when alarm values only are not sufficient to distinguish faults, and the final classification outcome is based on the fusion of the discrete and continuous classifiers.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Adnan, N. A., Izadi, I., & Chen, T. (2011). On expected detection delays for alarm systems with deadbands and delay-timers. *Journal of Process Control*, *21*, 1318–1331. http://dx.doi.org/10.1016/j.jprocont.2011.06.019.

Alcala, C. F., & Qin, S. J. (2009). Reconstruction-based contribution for process monitoring. *Automatica*, *45*, 1593–1600. http://dx.doi.org/10.1016/j.automatica.2009.02.027.

Ardakani, M., Askarian, M., Shokry, A., Escudero, G., Graells, M., & Espuña, A. (2016). Optimal features selection for designing a fault diagnosis system. *Computer Aided Chemical Engineering*, *38*, 1111–1116. http://dx.doi.org/10.1016/B978-0-444-63428-3.50190-9.

Chai, Z., & Zhao, C. (2019). Enhanced random forest with concurrent analysis of static and dynamic nodes for industrial fault classification. *IEEE Transactions on Industrial Informatics*, *PP*, http://dx.doi.org/10.1109/tii.2019.2915559.

Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. *Computers and Electrical Engineering*, *40*, 16–28. http://dx.doi.org/10.1016/j.compeleceng.2013.11.024, arXiv:1606.03476.

Chiang, L. H., Kotanchek, M. E., & Kordon, A. K. (2004). Fault diagnosis based on Fisher discriminant analysis and support vector machines. *Computers and Chemical Engineering*, *28*, 1389–1401. http://dx.doi.org/10.1016/j.compchemeng.2003.10.002.

Ghosh, K., Ramteke, M., & Srinivasan, R. (2014). Optimal variable selection for effective statistical process monitoring. *Computers and Chemical Engineering*, *60*, 260–276. http://dx.doi.org/10.1016/j.compchemeng.2013.09.014.

Gonzalez, R., & Huang, B. (2014). Control-loop diagnosis using continuous evidence through kernel density estimation. *Journal of Process Control*, *24*, 640–651. http://dx.doi.org/10.1016/j.jprocont.2014.03.005.

He, X. B., Wang, W., Yang, Y. P., & Yang, Y. H. (2009). Variable-weighted Fisher discriminant analysis for process fault diagnosis. *Journal of Process Control*, *19*, 923–931. http://dx.doi.org/10.1016/j.jprocont.2008.12.001.

Hoskins, J. C., Kaliyur, K. M., & Himmelblau, D. M. (1991). Fault diagnosis in complex chemical plants using artificial neural networks. *AIChE Journal*, *37*, 137–141. http://dx.doi.org/10.1002/aic.690370112.

Hu, W., Wang, J., Chen, T., & Shah, S. L. (2017). Cause-effect analysis of industrial alarm variables using transfer entropies. *Control Engineering Practice*, *64*, 205–214. http://dx.doi.org/10.1016/j.conengprac.2017.04.012.

Hu, Y., & Zhao, C. (2019). Fault diagnosis with dual cointegration analysis of common and specific nonstationary fault variations. *IEEE Transactions on Automation Science and Engineering*, *PP*, 1–11. http://dx.doi.org/10.1109/tase.2019.2917580.

Huang, B. (2007). Bayesian methods for control loop monitoring and diagnosis. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, *40*, 29–38. http://dx.doi.org/10.3182/20070606-3-MX-2915.00004.

IEC (International Electrotechnical Commission) (2014). *Management of alarm systems for the process industries. IEC 62682*. IEC.

Jiang, Q., Huang, B., Ding, S. X., & Yan, X. (2016a). Bayesian fault diagnosis with asynchronous measurements and its application in networked distributed monitoring. *IEEE Transactions on Industrial Electronics*, *63*, 6316–6324. http://dx.doi.org/10.1109/TIE.2016.2577545.

Jiang, Q., Huang, B., & Yan, X. (2016b). GMM and optimal principal components-based Bayesian method for multimode fault diagnosis. *Computers and Chemical Engineering*, *84*, 338–349. http://dx.doi.org/10.1016/j.compchemeng.2015.09.013.

John, G. H., & Langley, P. (1995). Estimating continuous distributions in Bayesian classifiers. In *Proceedings of the eleventh conference on uncertainty in artificial intelligence (UAI1995)*. http://dx.doi.org/10.1128/AAC.03728-14.

Kraskov, A., Stögbauer, H., & Grassberger, P. (2004). Estimating mutual information. *Physical Review E - Statistical Physics*, *69*, 1–16. http://dx.doi.org/10.1103/PhysRevE.69.066138.

Kuang, T. H., Yan, Z., & Yao, Y. (2015). Multivariate fault isolation via variable selection in discriminant analysis. *Journal of Process Control*, *35*, 30–40. http://dx.doi.org/10.1016/j.jprocont.2015.08.011.

Lucke, M., Chioua, M., Grimholt, C., Hollender, M., & Thornhill, N. F. (2018). On improving fault detection and diagnosis using alarm-range normalisation. In *Proceedings of 10th SAFEPROCESS symposium* (pp. 1227–1232). Warsaw, Poland: http://dx.doi.org/10.1016/j.ifacol.2018.09.695, IFAC-PapersOnLine.

Lucke, M., Mei, X., Stief, A., Chioua, M., & Thornhill, N. F. (2019). Variable selection for fault detection and identification based on mutual information of alarm series. In *Proceedings of 12th DYCOPS symposium* (pp. 673–678). Florianópolis, Brazil: http://dx.doi.org/10.1016/j.ifacol.2019.06.140, IFAC-PapersOnLine.

Ming, L., & Zhao, J. (2017). Review on chemical process fault detection and diagnosis. In *2017 6th international symposium on advanced control of industrial processes, AdCONIP 2017* (pp. 457–462). IEEE, http://dx.doi.org/10.1109/ADCONIP.2017.7983824.

Naghoosi, E., Izadi, I., & Chen, T. (2011). Estimation of alarm chattering. *Journal of Process Control*, *21*, 1243–1249. http://dx.doi.org/10.1016/j.jprocont.2011.07.015.

Parzen, E. (2007). On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, *33*, 1065–1076. http://dx.doi.org/10.1214/aoms/1177704472.

Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*, 1226–1238. http://dx.doi.org/10.1109/TPAMI.2005.159.

Peres, F. A. P., & Fogliatto, F. S. (2018). Variable selection methods in multivariate statistical process control: A systematic literature review. *Computers & Industrial Engineering*, *115*, 603–619. http://dx.doi.org/10.1016/j.cie.2017.12.006.

Ross, B. C. (2014). Mutual information between discrete and continuous data sets. *PLoS ONE*, *9*, e87357. http://dx.doi.org/10.1371/journal.pone.0087357.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, *27*, 379–423. http://dx.doi.org/10.1002/j.1538-7305.1948.tb01338.x, arXiv:9411012.

Shu, Y., Ming, L., Cheng, F., Zhang, Z., & Zhao, J. (2016). Abnormal situation management: Challenges and opportunities in the big data era. *Computers and Chemical Engineering*, *91*, 104–113. http://dx.doi.org/10.1016/j.compchemeng.2016.04.011.

Stief, A., Ottewill, J. R., Tan, R., & Cao, Y. (2018a). Process and alarm data integration under a two-stage Bayesian framework for fault diagnostics. *IFAC-PapersOnLine*, *51*, 1220–1226. http://dx.doi.org/10.1016/j.ifacol.2018.09.696.

Stief, A., Tan, R., Cao, Y., & Ottewill, J. R. (2018b). Analytics of heterogeneous process data: Multiphase flow facility case study. *IFAC-PapersOnLine*, *51*, 363–368. http://dx.doi.org/10.1016/j.ifacol.2018.09.327.

Su, J., Wang, D., Zhang, Y., Yang, F., Zhao, Y., & Pang, X. (2017). Capturing causality for fault diagnosis based on multi-valued alarm series using transfer entropy. *Entropy*, *19*(663), http://dx.doi.org/10.3390/e19120663.

Tidriri, K., Tiplica, T., Chatti, N., & Verron, S. (2018). A generic framework for decision fusion in fault detection and diagnosis. *Engineering Applications of Artificial Intelligence*, *71*, 73–86. http://dx.doi.org/10.1016/j.engappai.2018.02.014.

Tong, C., & Palazoglu, A. (2016). Dissimilarity-based fault diagnosis through ensemble filtering of informative variables. *Industrial and Engineering Chemistry Research*, *55*, 8774–8783. http://dx.doi.org/10.1021/acs.iecr.6b00915.

Verron, S., Tiplica, T., & Kobi, A. (2006). Fault diagnosis with bayesian networks: Application to the Tennessee Eastman process. In *Proceedings of the IEEE international conference on industrial technology* (pp. 98–103). IEEE, http://dx.doi.org/10.1109/ICIT.2006.372301.

Verron, S., Tiplica, T., & Kobi, A. (2008). Fault detection and identification with a new feature selection based on mutual information. *Journal of Process Control*, *18*, 479–490. http://dx.doi.org/10.1016/j.jprocont.2007.08.003.

Wand, M., Jones, M., & Jones, M. (1994). *Kernel smoothing*. Chapman and Hall/CRC, http://dx.doi.org/10.1201/b14876.

Wu, H., & Zhao, J. (2018). Deep convolutional neural network model based chemical process fault diagnosis. *Computers and Chemical Engineering*, *115*, 185–197. http://dx.doi.org/10.1016/j.compchemeng.2018.04.009.

Yin, S., Ding, S. X., Xie, X., & Luo, H. (2014). A review on basic data-driven approaches for industrial process monitoring. *IEEE Transactions on Industrial Electronics*, *61*, 6414–6428. http://dx.doi.org/10.1109/TIE.2014.2301773.

Yu, W., & Yang, F. (2015). Detection of causality between process variables based on industrial alarm data using transfer entropy. *Entropy*, *17*, 5868–5887. http://dx.doi.org/10.3390/e17085868.

Yu, W., & Zhao, C. (2019). Online fault diagnosis for industrial processes with Bayesian network-based probabilistic ensemble learning strategy. *IEEE Transactions on Automation Science and Engineering*, *PP*, 1–11. http://dx.doi.org/10.1109/tase.2019.2915286.

Zeng, Y., Jia, Z., Liang, W., & Gu, S. (2019). Fault diagnosis based on variable-weighted separability-oriented subclass discriminant analysis. *Computers and Chemical Engineering Chemical Engineering*, *129*, 106514. http://dx.doi.org/10.1016/j.compchemeng.2019.106514.

Zhang, Z., & Zhao, J. (2017). A deep belief network based fault diagnosis model for complex chemical processes. *Computers and Chemical Engineering*, *107*, 395–407. http://dx.doi.org/10.1016/j.compchemeng.2017.02.041.

Zhao, C., & Gao, F. (2017). Critical-to-fault-degradation variable analysis and direction extraction for online fault prognostic. *IEEE Transactions on Control Systems Technology*, *25*, 842–854. http://dx.doi.org/10.1109/TCST.2016.2576018.

Zhu, W., Sun, W., & Romagnoli, J. (2018). Adaptive k-nearest-neighbor method for process monitoring. *Industrial and Engineering Chemistry Research*, *57*, 2574–2586. http://dx.doi.org/10.1021/acs.iecr.7b03771.